CCPROG2 Machine Project Specifications
Term 2, AY 2024 – 2025

Building a Pokédex! Pokémon Stats Database and Battle Simulation

A Pokédex is a fictional electronic encyclopedia that contains detailed information about Pokémon. This handheld device is an essential tool for Pokémon trainers, allowing them to record and access data on the Pokémon they encounter during their adventures.

One day, you receive an urgent message from Professor Oak, the renowned Pokémon researcher. He has chosen you for a special mission: to build a revamped version of the Pokédex. This new Pokédex will be more advanced and feature-rich than ever before. Here are the exciting new features and functionalities you will incorporates:

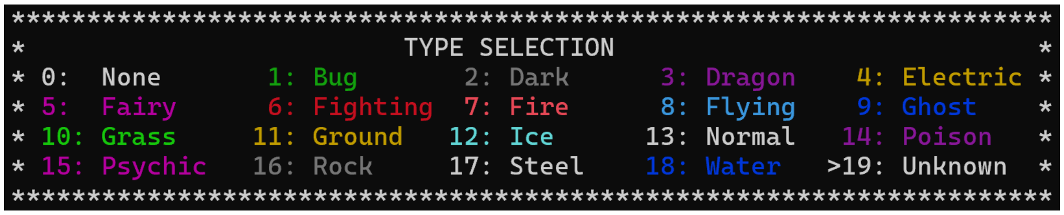
****

1. Show Pokémon Types
2. Add New Pokémon Entry
3. Display Complete Pokédex Chart
4. Display Complete Pokédex Table
5. Sort and Display Pokédex Table Based on Stats
6. Search Pokémon
7. Edit Pokémon
8. Delete Pokémon
9. Simulate Level Up
10. Simulate Battle
11. Display Battle Details
12. Exit

****

1. Show Pokémon Types

This functionality shows all the current Pokémon types there is. They are numbered from zero to 18 as indicated in the screenshot below. Entering a number beyond the specified scope corresponds to an "Unknown", "Undefined", or "N/A" type. A Pokémon can have at most two types.



```
**********************************************************************
*                           TYPE SELECTION                           *
* 0:  None       1:  Bug        2:  Dark       3:  Dragon    4:  Electric *
* 5:  Fairy      6:  Fighting   7:  Fire       8:  Flying    9:  Ghost    *
* 10: Grass      11: Ground     12: Ice        13: Normal    14: Poison   *
* 15: Psychic    16: Rock       17: Steel      18: Water     >19: Unknown *
**********************************************************************
```

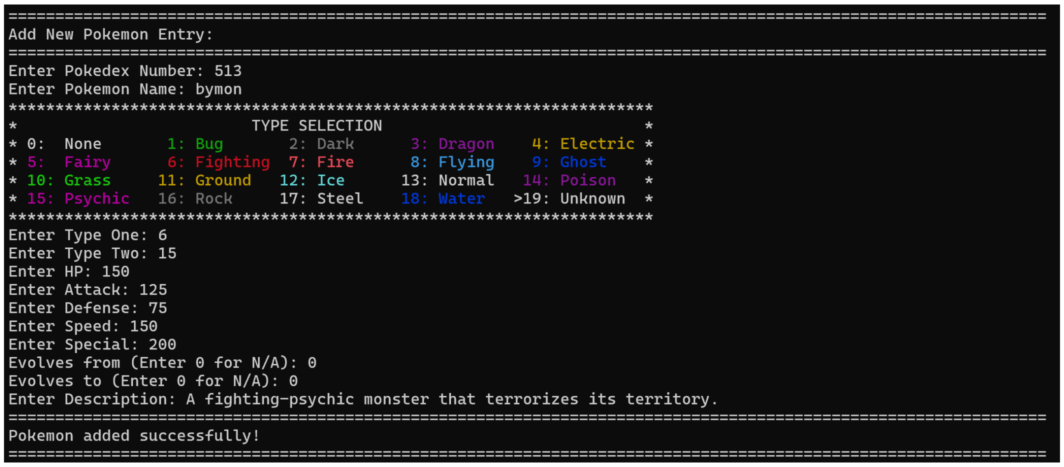
Figure 1 - Displaying all the Pokémon Types

2. Add a New Pokémon Entry

In adding a new Pokémon entry, the user inputs the following details:

a. Pokédex Number – must be a positive integer.
b. Pokémon Name – a string for the Pokémon name (Maximum string length of 30).
c. Type One – show the Pokémon Types and select among the options.
d. Type Two – show the Pokémon Types and select among the options. Take note that Type One should not be the same with Type Two. Prompt the user to re-enter Type Two.

e. HP – a positive integer.
f. Attack – a positive integer.
g. Defense – a positive integer.
h. Speed – a positive integer.
i. Special – a positive integer.
j. Evolves from – an integer that matches to another Pokémon if it already exists. Returns as “N/A” if there are no matches. An input of ‘0’ is an automatic “N/A”.
k. Evolves to – an integer that matches to another Pokémon if it already exists. Returns as “N/A” if there are no matches. An input of ‘0’ is an automatic “N/A”.
l. Description – a brief description of the Pokémon.



```
===============================================================================================================
Add New Pokemon Entry:
===============================================================================================================
Enter Pokedex Number: 513
Enter Pokemon Name: bymon
*********************************************************************
*                         TYPE SELECTION                            *
* 0:  None       1: Bug       2: Dark      3: Dragon    4: Electric *
* 5:  Fairy      6: Fighting  7: Fire      8: Flying    9: Ghost    *
* 10: Grass     11: Ground   12: Ice      13: Normal   14: Poison   *
* 15: Psychic   16: Rock     17: Steel    18: Water  >19: Unknown  *
*********************************************************************
Enter Type One: 6
Enter Type Two: 15
Enter HP: 150
Enter Attack: 125
Enter Defense: 75
Enter Speed: 150
Enter Special: 200
Evolves from (Enter 0 for N/A): 0
Evolves to (Enter 0 for N/A): 0
Enter Description: A fighting-psychic monster that terrorizes its territory.
===============================================================================================================
Pokemon added successfully!
===============================================================================================================
```

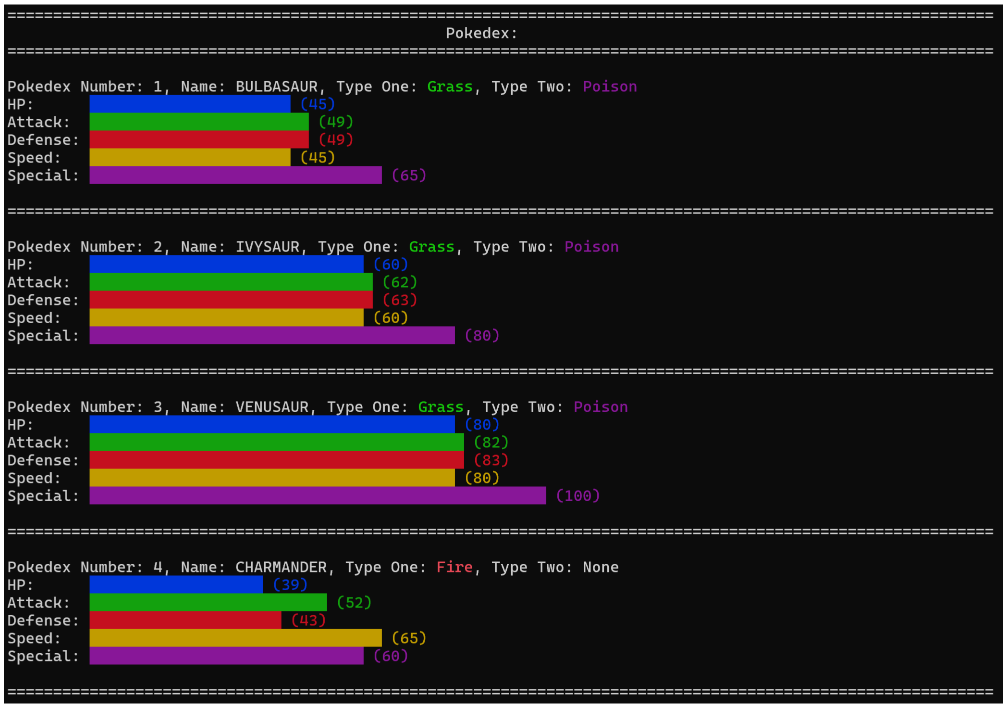
Figure 2 - Adding a New Pokémon Entry

RESTRICTIONS:

a) Pokédex number and Pokémon name must have no duplicates. If there are any duplicates, prompt the user to re-input the data. Duplicates in the names include those that are from different regions.
b) Stats must always be a positive integer.

3. Display Complete Pokédex Chart

This function simply shows all the Pokémons in the database with their stats shown as bars. The integer value must be shown at the end of each bar. The bars’ length is one half of the actual stat. For example, a Pokemon having 100 HP will have a bar the length of 50. Take note that the stat remains at 100 HP.



```
============================================================================================
                                         Pokedex:
============================================================================================

Pokedex Number: 1, Name: BULBASAUR, Type One: Grass, Type Two: Poison
HP:                          (45)
Attack:                        (49)
Defense:                       (49)
Speed:                       (45)
Special:                             (65)

============================================================================================

Pokedex Number: 2, Name: IVYSAUR, Type One: Grass, Type Two: Poison
HP:                                  (60)
Attack:                               (62)
Defense:                              (63)
Speed:                               (60)
Special:                                      (80)

============================================================================================

Pokedex Number: 3, Name: VENUSAUR, Type One: Grass, Type Two: Poison
HP:                                           (80)
Attack:                                        (82)
Defense:                                       (83)
Speed:                                        (80)
Special:                                                (100)

============================================================================================

Pokedex Number: 4, Name: CHARMANDER, Type One: Fire, Type Two: None
HP:                        (39)
Attack:                           (52)
Defense:                    (43)
Speed:                               (65)
Special:                           (60)

============================================================================================
```

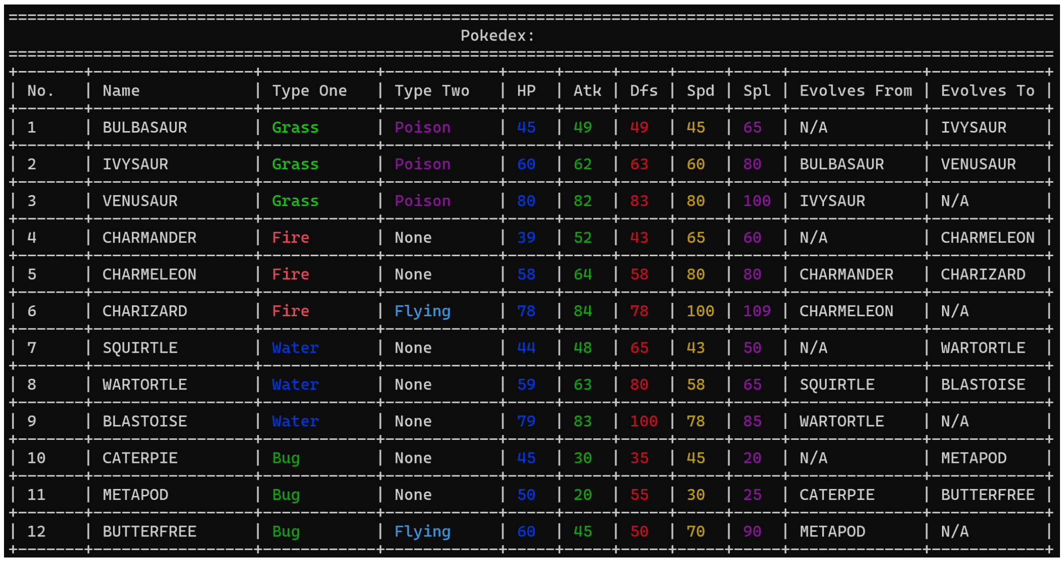
Figure 3 - Displaying the complete Pokédex Chart

OTHER SPECIFICATIONS:

a. Pokémons must be arranged in ascending order based on the Pokédex Number.
b. Pokémon descriptions are not required to be shown.
c. Evolves from and evolves to are not required to be shown.
d. Display proper spacing, alignment, and separators.

4. Display Complete Pokédex Table

This function shows all the Pokémons in the database but in a tabular form. As the same with the previous function, the description is not shown.



```
==================================================================================================
                                          Pokedex:
==================================================================================================
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| No.   | Name              | Type One   | Type Two   | HP  | Atk | Dfs | Spd | Spl | Evolves From | Evolves To  |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 1     | BULBASAUR         | Grass      | Poison     | 45  | 49  | 49  | 45  | 65  | N/A          | IVYSAUR     |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 2     | IVYSAUR           | Grass      | Poison     | 60  | 62  | 63  | 60  | 80  | BULBASAUR    | VENUSAUR    |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 3     | VENUSAUR          | Grass      | Poison     | 80  | 82  | 83  | 80  | 100 | IVYSAUR      | N/A         |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 4     | CHARMANDER        | Fire       | None       | 39  | 52  | 43  | 65  | 60  | N/A          | CHARMELEON  |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 5     | CHARMELEON        | Fire       | None       | 58  | 64  | 58  | 80  | 80  | CHARMANDER   | CHARIZARD   |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 6     | CHARIZARD         | Fire       | Flying     | 78  | 84  | 78  | 100 | 109 | CHARMELEON   | N/A         |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 7     | SQUIRTLE          | Water      | None       | 44  | 48  | 65  | 43  | 50  | N/A          | WARTORTLE   |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 8     | WARTORTLE         | Water      | None       | 59  | 63  | 80  | 58  | 65  | SQUIRTLE     | BLASTOISE   |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 9     | BLASTOISE         | Water      | None       | 79  | 83  | 100 | 78  | 85  | WARTORTLE    | N/A         |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 10    | CATERPIE          | Bug        | None       | 45  | 30  | 35  | 45  | 20  | N/A          | METAPOD     |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 11    | METAPOD           | Bug        | None       | 50  | 20  | 55  | 30  | 25  | CATERPIE     | BUTTERFREE  |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
| 12    | BUTTERFREE        | Bug        | Flying     | 60  | 45  | 50  | 70  | 90  | METAPOD      | N/A         |
+-------+-------------------+------------+------------+-----+-----+-----+-----+-----+--------------+-------------+
```

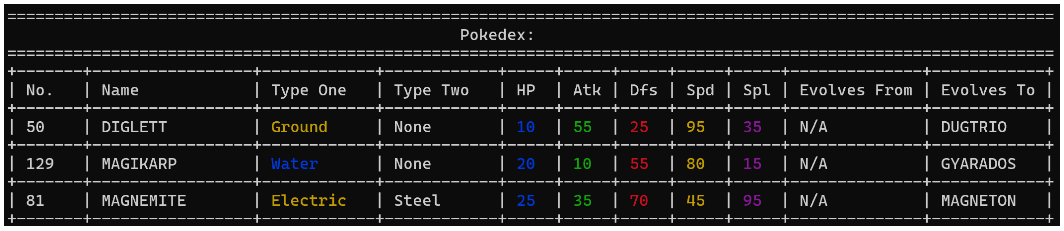
Figure 4 - Displaying the Complete Pokédex Table

OTHER SPECIFICATIONS:

e. Pokémons must be arranged in ascending order based on the Pokédex Number.
f. Pokémon descriptions are not required to be shown.
g. Display proper spacing, alignment, and separators.

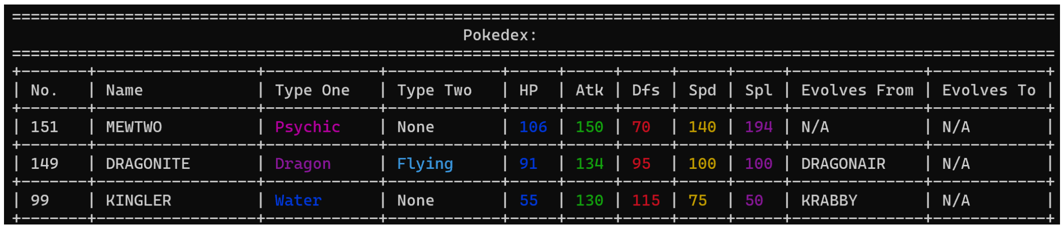
5. Sort and Display Pokédex Table Based on Stats

h. Sorting can be done with the following stats:
   i. HP
   ii. Attack
   iii. Defense
   iv. Speed
   v. Special
i. Sorting can either be Ascending or Descending



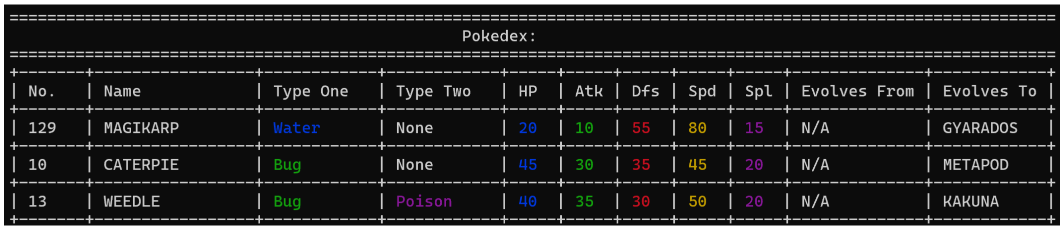
```
=================================================================================================================
                                                    Pokedex:
=================================================================================================================
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| No.   | Name               | Type One    | Type Two    | HP   | Atk  | Dfs  | Spd  | Spl  | Evolves From   | Evolves To  |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 50    | DIGLETT            | Ground      | None        | 10   | 55   | 25   | 95   | 35   | N/A            | DUGTRIO     |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 129   | MAGIKARP           | Water       | None        | 20   | 10   | 55   | 80   | 15   | N/A            | GYARADOS    |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 81    | MAGNEMITE          | Electric    | Steel       | 25   | 35   | 70   | 45   | 95   | N/A            | MAGNETON    |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
```

Figure 5 - HP: Ascending



```
=================================================================================================================
                                                    Pokedex:
=================================================================================================================
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| No.   | Name               | Type One    | Type Two    | HP   | Atk  | Dfs  | Spd  | Spl  | Evolves From   | Evolves To  |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 151   | MEWTWO             | Psychic     | None        | 106  | 150  | 70   | 140  | 194  | N/A            | N/A         |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 149   | DRAGONITE          | Dragon      | Flying      | 91   | 134  | 95   | 100  | 100  | DRAGONAIR      | N/A         |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 99    | KINGLER            | Water       | None        | 55   | 130  | 115  | 75   | 50   | KRABBY         | N/A         |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
```

Figure 6 - Attack: Descending



```
=================================================================================================================
                                                    Pokedex:
=================================================================================================================
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| No.   | Name               | Type One    | Type Two    | HP   | Atk  | Dfs  | Spd  | Spl  | Evolves From   | Evolves To  |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 129   | MAGIKARP           | Water       | None        | 20   | 10   | 55   | 80   | 15   | N/A            | GYARADOS    |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 10    | CATERPIE           | Bug         | None        | 45   | 30   | 35   | 45   | 20   | N/A            | METAPOD     |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
| 13    | WEEDLE             | Bug         | Poison      | 40   | 35   | 30   | 50   | 20   | N/A            | KAKUNA      |
+-------+--------------------+-------------+-------------+------+------+------+------+------+----------------+-------------+
```

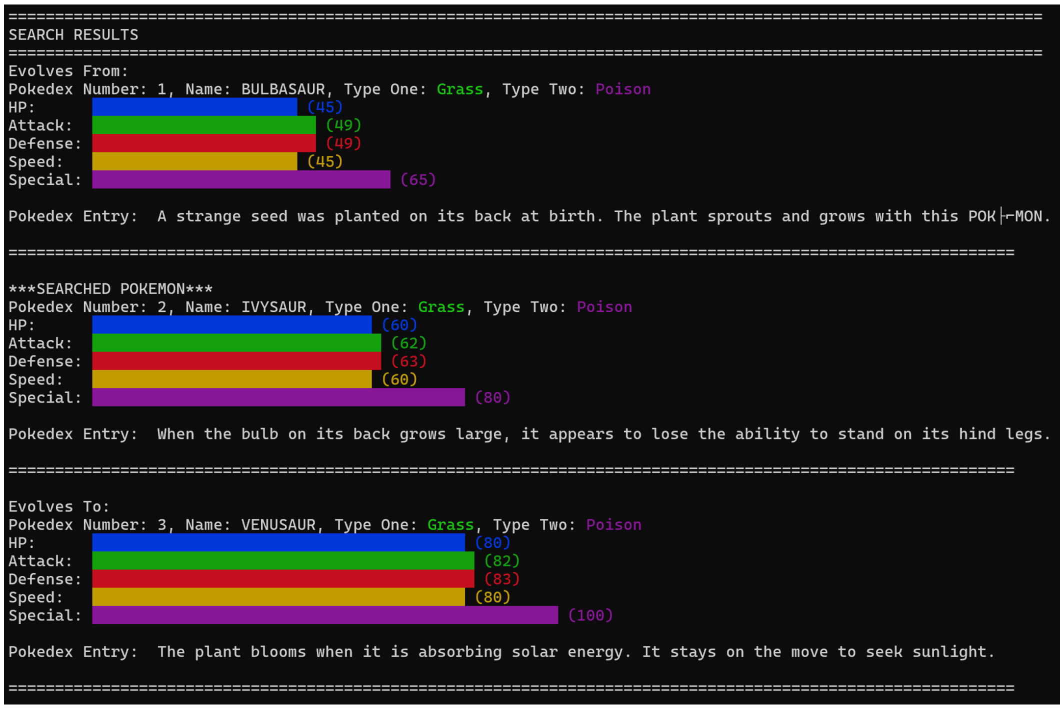
Figure 7 - Special: Ascending

6. Search Pokémon

This function allows the user to search the Pokémon database based on the following search terms:

j. Pokédex Number – an integer input
k. Pokémon Name – string input and is not case sensitive. Only complete names can be searched.
   Searching using methods ‘a’ and ‘b’ will yield the chart display. If a Pokémon has a valid “Evolves From” value, show it before the searched Pokémon. If a Pokémon has a valid

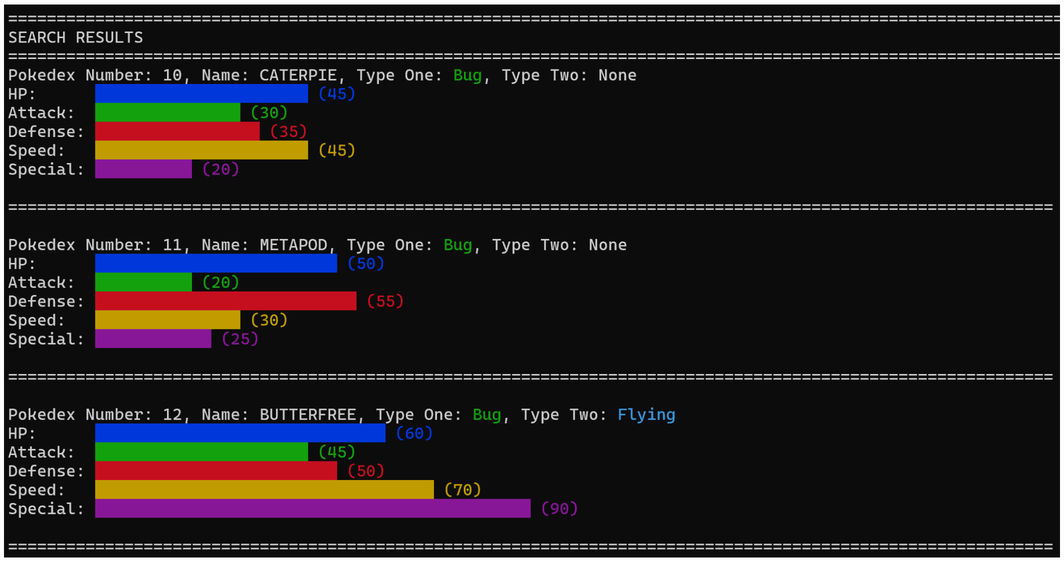
"Evolves To" value, show it after the searched Pokémon. This time, show all the Pokémons' descriptions as well.



```
==========================================================================================================
SEARCH RESULTS
==========================================================================================================
Evolves From:
Pokedex Number: 1, Name: BULBASAUR, Type One: Grass, Type Two: Poison
HP:                       (45)
Attack:                   (49)
Defense:                  (49)
Speed:                    (45)
Special:                  (65)

Pokedex Entry:  A strange seed was planted on its back at birth. The plant sprouts and grows with this POK├⌐MON.

==========================================================================================================

***SEARCHED POKEMON***
Pokedex Number: 2, Name: IVYSAUR, Type One: Grass, Type Two: Poison
HP:                       (60)
Attack:                   (62)
Defense:                  (63)
Speed:                    (60)
Special:                  (80)

Pokedex Entry:  When the bulb on its back grows large, it appears to lose the ability to stand on its hind legs.

==========================================================================================================

Evolves To:
Pokedex Number: 3, Name: VENUSAUR, Type One: Grass, Type Two: Poison
HP:                       (80)
Attack:                   (82)
Defense:                  (83)
Speed:                    (80)
Special:                  (100)

Pokedex Entry:  The plant blooms when it is absorbing solar energy. It stays on the move to seek sunlight.

==========================================================================================================
```

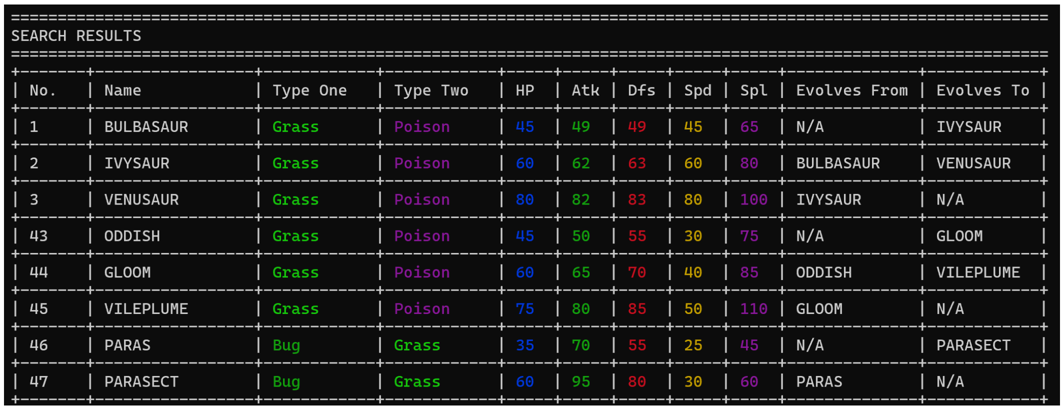
Figure 8 - Output if Pokédex Number '2' or Pokémon Name "Ivysaur" is searched

l. Type – this can search from either Type One or Type Two.
   i. Show the Pokémon Types for Reference
   ii. Search via type will show all the Pokémons that has a match with the search term and either Type One or Type Two
   iii. There is also an option to show the results via chart or table.



```
==========================================================================================================
SEARCH RESULTS
==========================================================================================================
Pokedex Number: 10, Name: CATERPIE, Type One: Bug, Type Two: None
HP:                       (45)
Attack:                   (30)
Defense:                  (35)
Speed:                    (45)
Special:                  (20)

==========================================================================================================

Pokedex Number: 11, Name: METAPOD, Type One: Bug, Type Two: None
HP:                       (50)
Attack:                   (20)
Defense:                  (55)
Speed:                    (30)
Special:                  (25)

==========================================================================================================

Pokedex Number: 12, Name: BUTTERFREE, Type One: Bug, Type Two: Flying
HP:                       (60)
Attack:                   (45)
Defense:                  (50)
Speed:                    (70)
Special:                  (90)

==========================================================================================================
```

Figure 9 - Searching for Type "Bug" and display the chart

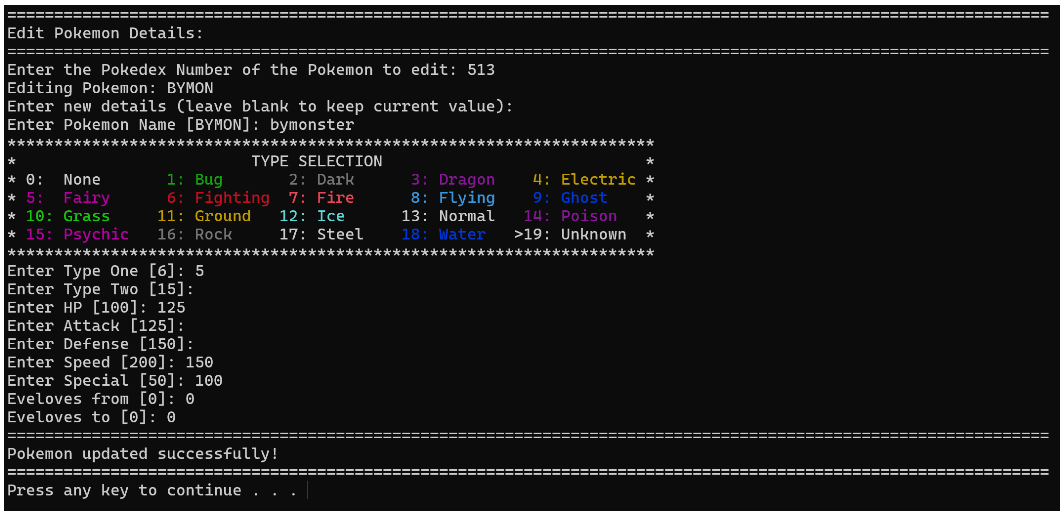


```
=================================================================================================================
SEARCH RESULTS
=================================================================================================================
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| No.   | Name              | Type One   | Type Two   | HP   | Atk  | Dfs  | Spd  | Spl  | Evolves From | Evolves To   |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 1     | BULBASAUR         | Grass      | Poison     | 45   | 49   | 49   | 45   | 65   | N/A          | IVYSAUR      |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 2     | IVYSAUR           | Grass      | Poison     | 60   | 62   | 63   | 60   | 80   | BULBASAUR    | VENUSAUR     |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 3     | VENUSAUR          | Grass      | Poison     | 80   | 82   | 83   | 80   | 100  | IVYSAUR      | N/A          |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 43    | ODDISH            | Grass      | Poison     | 45   | 50   | 55   | 30   | 75   | N/A          | GLOOM        |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 44    | GLOOM             | Grass      | Poison     | 60   | 65   | 70   | 40   | 85   | ODDISH       | VILEPLUME    |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 45    | VILEPLUME         | Grass      | Poison     | 75   | 80   | 85   | 50   | 110  | GLOOM        | N/A          |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 46    | PARAS             | Bug        | Grass      | 35   | 70   | 55   | 25   | 45   | N/A          | PARASECT     |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
| 47    | PARASECT          | Bug        | Grass      | 60   | 95   | 80   | 30   | 60   | PARAS        | N/A          |
+-------+-------------------+------------+------------+------+------+------+------+------+--------------+--------------+
```

Figure 10 - Searching for "Grass" and displaying the table

7. Edit Pokémon

Editing a Pokémon allows the user to change all of its attributes. To access the Pokémon, the user needs to input its Pokédex Number; which is a value that cannot be edited. In this function, the user is presented with the current value of each attribute. If there are no changes, the user may simply hit "Enter" to retain the value. If there are changes, simply input the new value. The description should also not be edited.

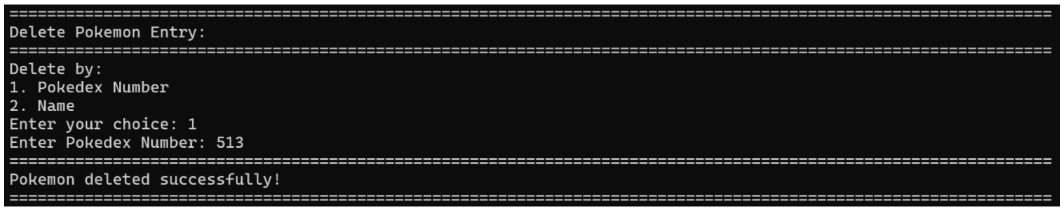


```
=================================================================================================================
Edit Pokemon Details:
=================================================================================================================
Enter the Pokedex Number of the Pokemon to edit: 513
Editing Pokemon: BYMON
Enter new details (leave blank to keep current value):
Enter Pokemon Name [BYMON]: bymonster
*********************************************************************
*                          TYPE SELECTION                           *
* 0:  None       1: Bug        2: Dark       3: Dragon    4: Electric *
* 5:  Fairy      6: Fighting   7: Fire       8: Flying    9: Ghost    *
* 10: Grass     11: Ground    12: Ice       13: Normal   14: Poison   *
* 15: Psychic   16: Rock      17: Steel     18: Water   >19: Unknown  *
*********************************************************************
Enter Type One [6]: 5
Enter Type Two [15]:
Enter HP [100]: 125
Enter Attack [125]:
Enter Defense [150]:
Enter Speed [200]: 150
Enter Special [50]: 100
Eveloves from [0]: 0
Eveloves to [0]: 0
=================================================================================================================
Pokemon updated successfully!
=================================================================================================================
Press any key to continue . . .
```

Figure 11 - Editing Pokémon Details

8. Delete Pokémon

This function allows the user to delete a Pokémon from the Pokédex. The user has the option to delete using the Pokédex Number or the Pokémon Name (not case sensitive). If the Pokémon is found, it will be deleted from the database.

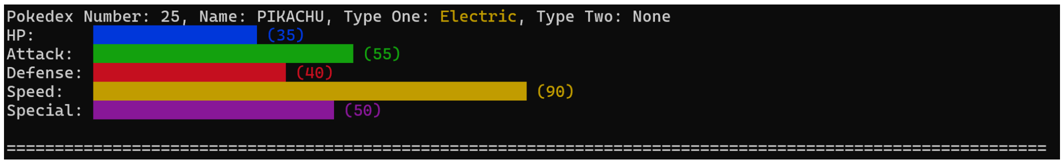


```
=================================================================================================================
Delete Pokemon Entry:
=================================================================================================================
Delete by:
1. Pokedex Number
2. Name
Enter your choice: 1
Enter Pokedex Number: 513
=================================================================================================================
Pokemon deleted successfully!
=================================================================================================================
```

Figure 12 - Successful Deletion of a Pokémon

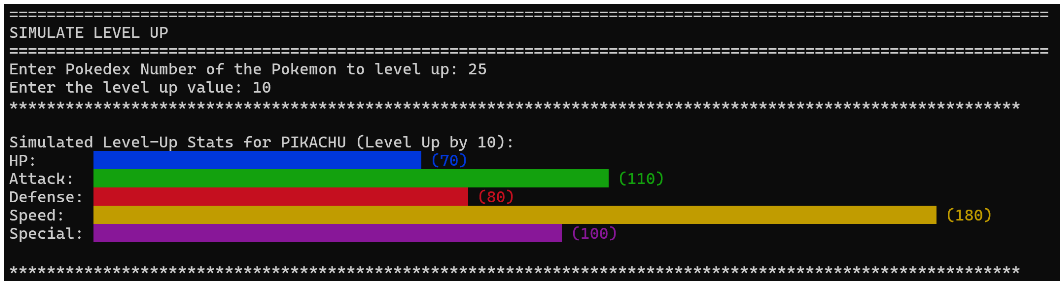
9. Simulate Level Up

This function simulates the stats of a Pokémon at a given level. Each level add 10% increase to all stats based on the base statistics. Take note to consider only the integer part of the increased stat; anything after the decimal point is disregarded.If a Pokémon has a base statistics of 50 and a level up value of 10, the formula is 50 + (50 * 0.10 * 10). Note that this is a simulation and should not update the records from the database.



```
Pokedex Number: 25, Name: PIKACHU, Type One: Electric, Type Two: None
HP:                        (35)
Attack:                              (55)
Defense:                      (40)
Speed:                                                 (90)
Special:                           (50)

============================================================================================================
```

Figure 13 - Pikachu's Original Base Stats



```
===============================================================================================================
SIMULATE LEVEL UP
===============================================================================================================
Enter Pokedex Number of the Pokemon to level up: 25
Enter the level up value: 10
************************************************************************************************************

Simulated Level-Up Stats for PIKACHU (Level Up by 10):
HP:                                          (70)
Attack:                                                          (110)
Defense:                                          (80)
Speed:                                                                                              (180)
Special:                                                    (100)

************************************************************************************************************
```

Figure 14 - Simulating Pikachu at Level 10

10. Simulate Battle

a) Simulating a battle matches the individual statistics of two Pokémon based on their base stats. Before tallying the stat points, a comparison of the Pokémon types must be made. Each type has an advantage or weakness against other types. In this scenario, the function will only consider the first type or Type One of the Pokémon. If the first Pokémon has a type advantage over the second, the disadvantaged Pokémon will have a 20% reduction to their base stats and vice versa.

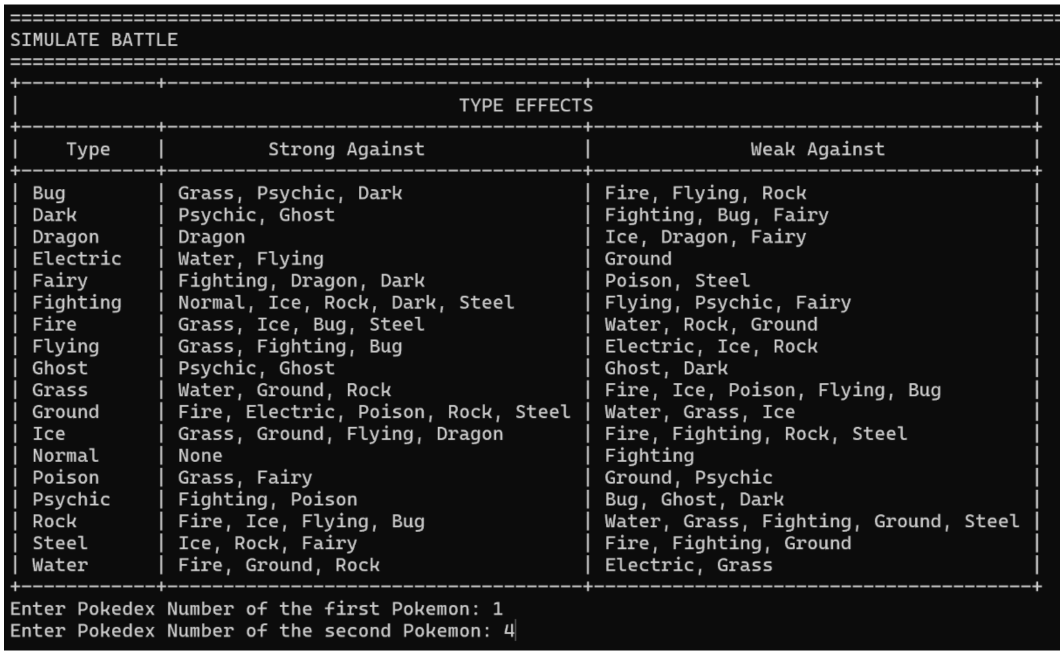
b) A scoring mechanism will check which of the two Pokémon is stronger. For the HP, Attack, Defense, Speed, and Special, one point will be awarded to whichever Pokémon has a higher statistic; no point will be awarded if the stats are the same.

c) A weighted average of all the previous stats will also be computed. It will merit two points for the Pokémon with the better average. The weights are 10% for the HP, 30% for the Attack, 20% for the Defense, 15% for the Speed, and 15% for the Special.

d) A maximum of seven points can be awarded.

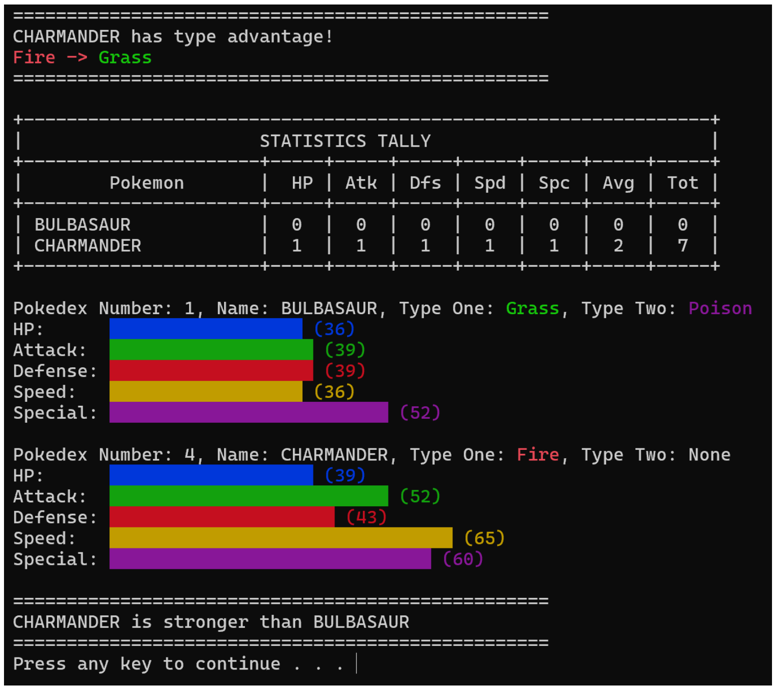
e) Based on the tally, the function must determine which of the two Pokémon is stronger; otherwise, the function will indicate a tie. In displaying the battle simulation, include all the details shown in the succeeding figures.

f) All battles must be recorded in a text file.



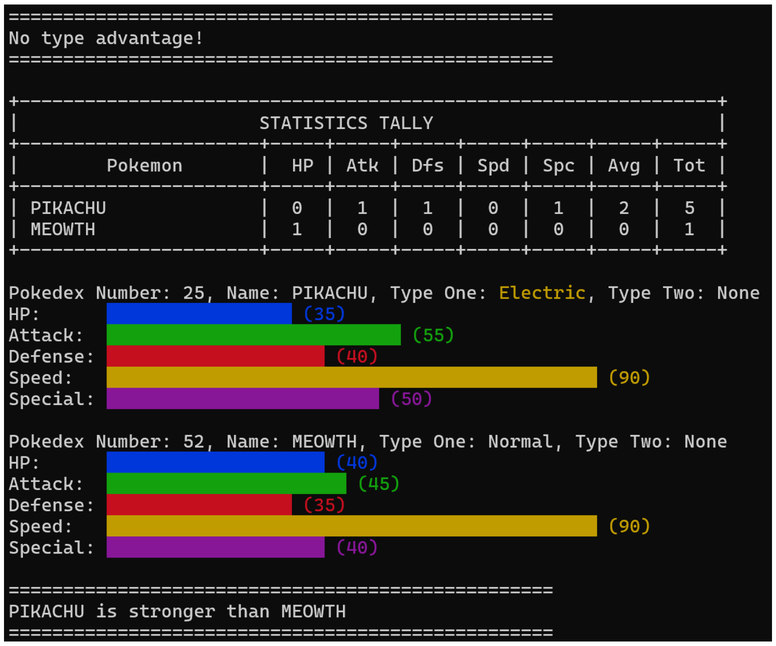
```
==============================================================================================
SIMULATE BATTLE
==============================================================================================
+-------------+--------------------------------------+-------------------------------------------+
|                                         TYPE EFFECTS                                            |
+-------------+--------------------------------------+-------------------------------------------+
|    Type     |           Strong Against             |               Weak Against                |
+-------------+--------------------------------------+-------------------------------------------+
| Bug         | Grass, Psychic, Dark                 | Fire, Flying, Rock                        |
| Dark        | Psychic, Ghost                       | Fighting, Bug, Fairy                      |
| Dragon      | Dragon                               | Ice, Dragon, Fairy                        |
| Electric    | Water, Flying                        | Ground                                    |
| Fairy       | Fighting, Dragon, Dark               | Poison, Steel                             |
| Fighting    | Normal, Ice, Rock, Dark, Steel       | Flying, Psychic, Fairy                    |
| Fire        | Grass, Ice, Bug, Steel               | Water, Rock, Ground                       |
| Flying      | Grass, Fighting, Bug                 | Electric, Ice, Rock                       |
| Ghost       | Psychic, Ghost                       | Ghost, Dark                               |
| Grass       | Water, Ground, Rock                  | Fire, Ice, Poison, Flying, Bug            |
| Ground      | Fire, Electric, Poison, Rock, Steel  | Water, Grass, Ice                         |
| Ice         | Grass, Ground, Flying, Dragon        | Fire, Fighting, Rock, Steel               |
| Normal      | None                                 | Fighting                                  |
| Poison      | Grass, Fairy                         | Ground, Psychic                           |
| Psychic     | Fighting, Poison                     | Bug, Ghost, Dark                          |
| Rock        | Fire, Ice, Flying, Bug               | Water, Grass, Fighting, Ground, Steel     |
| Steel       | Ice, Rock, Fairy                     | Fire, Fighting, Ground                    |
| Water       | Fire, Ground, Rock                   | Electric, Grass                           |
+-------------+--------------------------------------+-------------------------------------------+
Enter Pokedex Number of the first Pokemon: 1
Enter Pokedex Number of the second Pokemon: 4
```

Figure 15 - Type Effects



```
==================================================
CHARMANDER has type advantage!
Fire -> Grass
==================================================

+-----------------------------------------------------------------+
|                     STATISTICS TALLY                            |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
|        Pokemon        | HP  | Atk | Dfs | Spd | Spc | Avg | Tot |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
| BULBASAUR             |  0  |  0  |  0  |  0  |  0  |  0  |  0  |
| CHARMANDER            |  1  |  1  |  1  |  1  |  1  |  2  |  7  |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+

Pokedex Number: 1, Name: BULBASAUR, Type One: Grass, Type Two: Poison
HP:                        (36)
Attack:                     (39)
Defense:                    (39)
Speed:                     (36)
Special:                           (52)

Pokedex Number: 4, Name: CHARMANDER, Type One: Fire, Type Two: None
HP:                         (39)
Attack:                            (52)
Defense:                      (43)
Speed:                                   (65)
Special:                                (60)

==================================================
CHARMANDER is stronger than BULBASAUR
==================================================
Press any key to continue . . .
```

Figure 16 - One Pokémon has Type Advantage

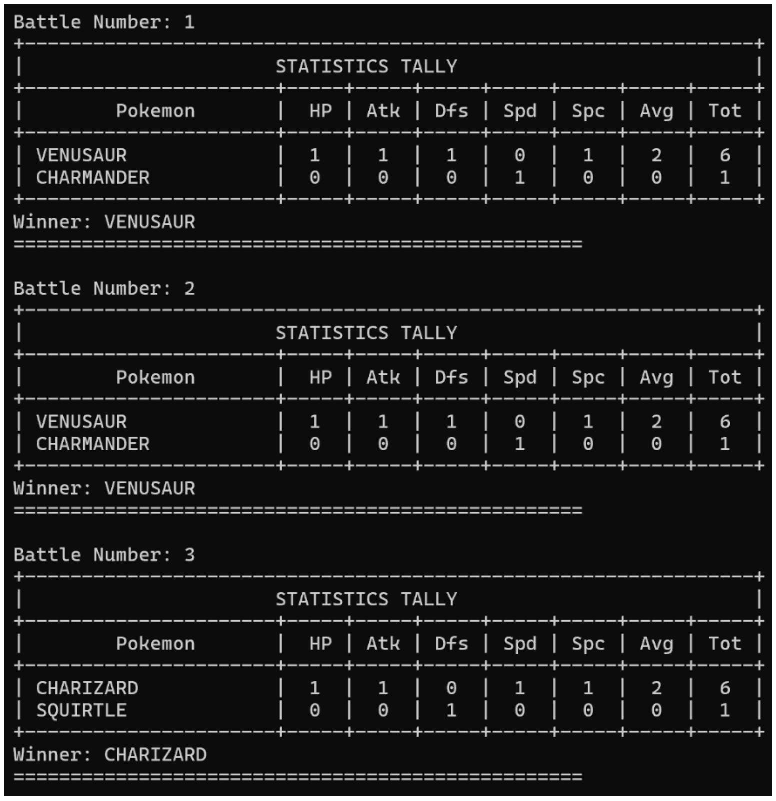


```
==================================================
No type advantage!
==================================================

+-----------------------------------------------------------------+
|                      STATISTICS TALLY                            |
+----------------------+-----+-----+-----+-----+-----+-----+-----+
|        Pokemon       |  HP | Atk | Dfs | Spd | Spc | Avg | Tot |
+----------------------+-----+-----+-----+-----+-----+-----+-----+
| PIKACHU              |  0  |  1  |  1  |  0  |  1  |  2  |  5  |
| MEOWTH               |  1  |  0  |  0  |  0  |  0  |  0  |  1  |
+----------------------+-----+-----+-----+-----+-----+-----+-----+

Pokedex Number: 25, Name: PIKACHU, Type One: Electric, Type Two: None
HP:                  (35)
Attack:                        (55)
Defense:                (40)
Speed:                                      (90)
Special:                     (50)

Pokedex Number: 52, Name: MEOWTH, Type One: Normal, Type Two: None
HP:                     (40)
Attack:                   (45)
Defense:             (35)
Speed:                                      (90)
Special:                (40)

==================================================
PIKACHU is stronger than MEOWTH
==================================================
```

Figure 17 - No Type Advantage

11. Display Battle Details

This function simply shows all the simulated battles indicating the summary.



```
Battle Number: 1
+-------------------------------------------------------------------+
|                       STATISTICS TALLY                            |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
|        Pokemon        |  HP | Atk | Dfs | Spd | Spc | Avg | Tot |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
| VENUSAUR              |  1  |  1  |  1  |  0  |  1  |  2  |  6  |
| CHARMANDER            |  0  |  0  |  0  |  1  |  0  |  0  |  1  |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
Winner: VENUSAUR
==================================================

Battle Number: 2
+-------------------------------------------------------------------+
|                       STATISTICS TALLY                            |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
|        Pokemon        |  HP | Atk | Dfs | Spd | Spc | Avg | Tot |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
| VENUSAUR              |  1  |  1  |  1  |  0  |  1  |  2  |  6  |
| CHARMANDER            |  0  |  0  |  0  |  1  |  0  |  0  |  1  |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
Winner: VENUSAUR
==================================================

Battle Number: 3
+-------------------------------------------------------------------+
|                       STATISTICS TALLY                            |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
|        Pokemon        |  HP | Atk | Dfs | Spd | Spc | Avg | Tot |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
| CHARIZARD             |  1  |  1  |  0  |  1  |  1  |  2  |  6  |
| SQUIRTLE              |  0  |  0  |  1  |  0  |  0  |  0  |  1  |
+-----------------------+-----+-----+-----+-----+-----+-----+-----+
Winner: CHARIZARD
==================================================
```

Figure 18 - Battle Records

12. Exit

Terminates the Program.

Strict Project Specifications and Implementation

1. This project can be accomplished individually or by pair. If you will choose the latter, make sure that both members of the pair contribute to the completion of the machine project.
2. The following concepts must strictly be used and enforced in the machine project:
    a. Functions
    b. Arrays
    c. Structs
    d. Sorting Algorithms
    e. Searching Algorithms
    f. File Management
        i. You are free to use any format in your text files. You may also use as many text files as you wish. Doing this allows you to fully customize your project while providing better optimization.
    g. Access to the text files in the form of saving to and loading from happens every time a functionality is executed. For example, for the "Add New Pokemon Entry" functionality, it will both read from and write to the same file. Other functionalities such as the "Show Pokemon Types" will only read the file.
3. Optional:
    a. Use of colors
4. Format your output based on your preference. Just ensure that the output is presented well. You may clear your screen to enhance the user experience. Do not use "sleep" or "pause" to make checking efficient.
5. Follow the sample outputs especially when it comes to the chart and the table. Spacing and alignment may differ a little but ensure that the presentation is optimized. Everyone is encouraged to write their own versions of the output to ensure that there is a variety among the submissions.

Other Project Details

1. A demonstration of the project will be shared with the class.
2. A sample spreadsheet of the Pokémon details will be shared with the class.
3. During demonstration, it is expected that the program already has a significant number of entries.

How to Approach the Machine Project

Step 1: Problem analysis and algorithm formulation

Read the MP Specifications again! Identify clearly what are the required information from the user, what kind of processes are needed, and what will be the output(s) of your program. Clarify with your professor any issues that you might have regarding the machine project.

When you have all the necessary information, identify the necessary functions that you will need to modularize the project. Identify the required data of these functions and what kind of data they will return to the caller. Write your algorithm for each of these modules/functions as well as the algorithm for your main program.

Step 2: Implementation

In this step, you are to translate your algorithm into proper C statements. While implementing, you are to perform the other phases of program planning and design (discussed in the other steps below) together with this step.

Follow the coding standard indicated in the course notes (Modules section in AnimoSpace).

You may choose to type your program in a text editor or an IDE (i.e. Dev-C IDE) at this point. Note that you are expected to use statements taught in class. You can explore other libraries and functions in C as long as you can clearly explain how this works. You may also use arrays, should these be applicable and you are able to properly justify and explain your implementation using these. For topics not covered, it is left to the student to read ahead, research, and explore by himself.

Note though that you are **NOT ALLOWED** to do the following:

- to declare and use global variables (i.e., variables declared outside any function),
- to use `goto` statements (i.e., to jump from code segments to code segments),
- to use the `break` statement to exit a block other than `switch` blocks,
- to use the `return` statement or `exit` statement to **prematurely** terminate a loop or function or program,
- to use the `exit` statement to **prematurely** terminate a loop or to terminate the function or program, and
- to call the `main()` function to repeat the process instead of using loops.

It is best that you perform your coding "incrementally." This means:

- dividing the program specification into subproblems, and solving each problem separately according to your algorithm;
- coding the solutions to the subproblems one at a time. Once you're done coding the solution for one subproblem, apply testing and debugging.

Documentation

While coding, you have to include internal documentation in your programs. You are expected to have the following:

- File comments or Introductory comments
- Function comments
- In-line comments

Introductory comments are found at the very beginning of your program before the preprocessor directives. Follow the format shown below. Note that items in between **< >** should be replaced with the proper information. Items in between ***[ ]*** are optional, indicate if applicable.

```
/*
    Description: <Describe what this program does briefly>
    Programmed by: <your name here> <section>
```

```
     Last modified: <date when last revision was made>
     Version: <version number>
     [Acknowledgements: <list of sites or borrowed libraries and
sources>]
*/
<Preprocessor directives>
<function implementation>
int main()
{
     return 0;
}
```

Function comments precede the function header. These are used to describe what the function does and the intentions of each parameter and what is being returned, if any. If applicable, include pre-conditions as well. Pre-conditions refer to the assumed state of the parameters. Follow the format below when writing function comments:

```
/*   <Description of function>
     Precondition: <precondition / assumption>
     @param <name> <purpose>
     @return <description of returned result>
*/
<return type> <function name> (<parameter list>);
```

Example:

```
/* This function computes for the area of a triangle
   Precondition: base and height are non-negative values
   @param base is the base measurement of the triangle in cm
   @param height is the height measurement of the triangle in
   cm
   @return the resulting area of the triangle
*/
float
getAreaTri (float base, float height)
{
    ...
}
```

In-Line comments are other comments in major parts of the code. These are expected to explain the purpose or algorithm of groups of related code, esp. for long functions.

Step 3: Testing and Debugging

SUBMIT THE LIST OF TEST CASES YOU HAVE USED.

For each feature of your program, you have to fully test it before moving to the next feature. Sample questions that you should ask yourself are:

1) What should be displayed on the screen after a user input?
2) What would happen if inputs are incorrect? (e.g., values not within the range)
3) Is my program displaying the correct output?
4) Is my program following the correct sequence of events (correct program flow)?
5) Is my program terminating (ending/exiting) correctly? Does it exit when I press the command to quit? Does it exit when the program's goal has been met? Is there an infinite loop?
6) and others...

IMPORTANT POINTS TO REMEMBER:

1. You are required to implement the project using the C language (C99 and NOT C++). Make sure you know how to compile and run in both the IDE (DEV-C++) and the command prompt (via
   `gcc –Wall -std=c99 <yourMP.c> -o <yourExe.exe>`
2. The implementation will require you to:
   a. Create and Use Functions
   b. **Note:** Non-use of self-defined functions will merit a grade of **0** for the **machine project**. Too few self-defined functions may merit deductions. A general rule is to create a separate function for each option described above, unless some features are too similar that one function can serve the purpose for two [or more] of the options. Note that functions whose tasks are only to display are not included in the count for creating user-defined functions.
   c. Appropriately use conditional statements, loops and other constructs discussed in class (Do not use brute force solution. **You are not allowed to use goto label statements, exit statements. You are required to pass parameters to functions and not allowed to declare global or static variables.**) Refer to Step 2 on Implementation for other details and restrictions.
      i. Consistently employ coding conventions
   d. Include internal documentation (i.e., comments)
3. Milestones:
   a. Milestone 1 covers functions one to four. Submission is until March 6, 2025 @ 11:59PM. This submission will constitute 30% of your MP grade.
   b. Milestone 2 covers all functions as indicated in the MP specifications. The deadline for this milestone is indicated in the succeeding number. This submission will constitute 70% of your MP grade. However, non-submission will invalidate the grade of the first milestone.
4. Deadline for the project (Milestone 2) is **7:30AM of March 31, 2025 (Monday)** via submission through **AnimoSpace**. Submissions from **7:30AM of March 31, 2025** will not be accepted anymore as the facility is locked. Once locked, no MP will be accepted anymore and this will result in a **0.0** for your machine project.
5. The following are the deliverables:

Checklist:

- Upload in **AnimoSpace** by clicking **Submit Assignment** on Machine Project and adding a zipped file with the following contents:
  - source code*
  - test script**
  - text files
- email the softcopies of everything as attachments to YOUR own email address on or before the deadline

Legend:

* Source Code also includes the internal documentation. The **first few lines of the source code** should have the following declaration (in comment) **BEFORE** the introductory comment:

```
/*****************************************************************
******
This is to certify that this project is my own work, based on
my personal efforts in studying and applying the concepts
learned. I have constructed the functions and their respective
algorithms
and corresponding code by myself. The program was run, tested,
and debugged by my own efforts. I further certify that I have
not copied in part or whole or otherwise plagiarized the work
of other students and/or persons.

<your full name>, DLSU ID# <number>
*****************************************************************
*****/
```

** Test Script should be in a table format, with header as shown below. There should be at least 3 distinct test classes (as indicated in the description) **per function**. There is no need to test functions which are only for screen design.

Sample is shown below.

Function	**#**	**Description**	**Sample Input Data**	**Expected Output**	**Actual Output**	**P/F**
sortIncreasing	1	Integers in array are in increasing order already	aData contains: 1 3 7 8 10 15 32 33 37 53	aData contains: 1 3 7 8 10 15 32 33 37 53	aData contains: 1 3 7 8 10 15 32 33 37 53	P
	2	Integers in array are in decreasing order	aData contains: 53 37 33 32 15 10 8 7 3 1	aData contains: 1 3 7 8 10 15 32 33 37 53	aData contains: 1 3 7 8 10 15 32 33 37 53	P
	3	Integers in array are combination of positive and negative numbers	aData contains: 57 30 -4 6 -5 -33 -96 0 82 -1	aData contains: -96 -33 -5 -4 -1 0 6 30 57 82	aData contains: -96 -33 -5 -4 -	P

		and in no particular sequence			1 0 6 30 57 82	

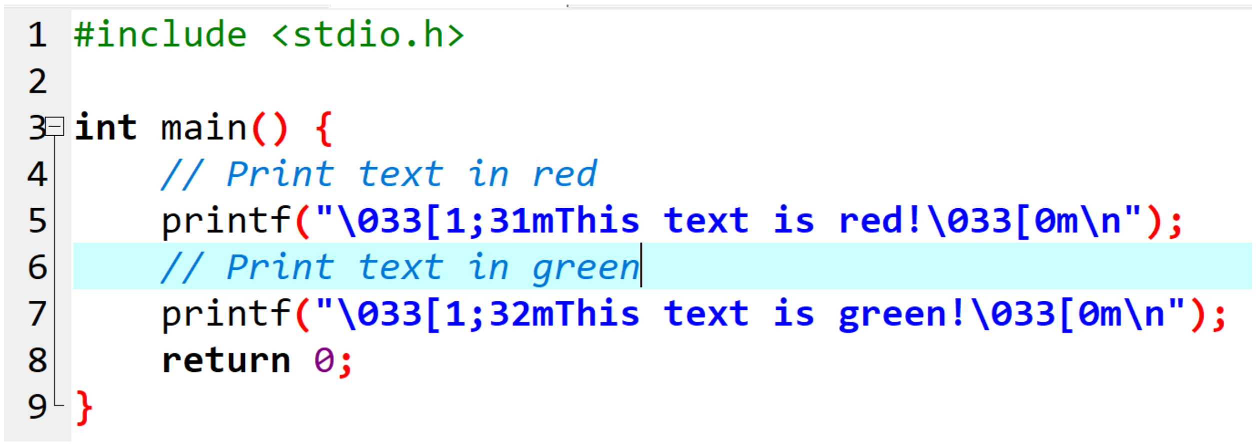
6. **MP Demo:** You will demonstrate your project on a specified schedule during the last weeks of classes. Being unable to show up on time during the demo or being unable to answer the questions convincingly during the demo will merit a grade of 0.0 for the **MP**. Both members of the group must be familiar with the code. They may assign among themselves which function to contribute with but ultimately, both members must know the entirety of their code. The project is initially evaluated via black box testing (i.e., based on output of the running program). Thus, if the program does not compile successfully using `gcc -Wall -std=c99` and executes in the command prompt, a grade of 0 for the project will be incurred. However, a fully working project does not ensure a perfect grade, as the implementation (i.e., correctness and compliance in code) is still checked.
7. Any requirement not fully implemented, and instruction not followed will merit deductions.
8. This project can be done individually or by pair. Working in collaboration, asking other people's/pair's help, borrowing or copying other people's/pair's work or from books or online sources (either in full or in part) are considered cheating. Cheating is punishable by a grade of **0.0** for **CCPROG2** course. Aside from which, a cheating case may be filed with the Discipline Office.
9. **Bonus points:** A maximum of 10 points may be given for features over & above the requirements, like an option to create and view ASCII arts of Pokemons, creating a "Ranking" of the strongest Pokemons based on stats against all others in the Pokedex, or creating an evolution tree for a Pokemon line; or other features not conflicting with the given requirements or changing the requirements) subject to evaluation of the teacher. Required features must be completed first before bonus features are credited. Note that use of conio.h, or other advanced C commands/statements may not necessarily merit bonuses.

HONESTY POLICY AND INTELLECTUAL PROPERTY RIGHTS

Honesty policy applies. Please take note that you are NOT allowed to borrow and/or copy-and-paste – in full or in part any existing related program code from the internet or other sources (such as printed materials like books, or source codes by other people that are not online). **You should develop your own codes from scratch by yourself.**

Appendix

To output text in color in C, you can use ANSI escape codes. These codes are special sequences that the terminal interprets to change text color and other formatting options. Here's a simple example:



```c
#include <stdio.h>

int main() {
    // Print text in red
    printf("\033[1;31mThis text is red!\033[0m\n");
    // Print text in green
    printf("\033[1;32mThis text is green!\033[0m\n");
    return 0;
}
```

In this example:

- \033[1;31m sets the text color to red.
- \033[1;32m sets the text color to green.
- \033[0m resets the text color to default.

Here are some common color codes:

- Black: 30
- Red: 31
- Green: 32
- Yellow: 33
- Blue: 34
- Magenta: 35
- Cyan: 36
- White: 37